

1. Title of the Invention

Microcomputer

2. Claims

1. A microcomputer having an electrically writable ROM, comprising:

the electrically writable ROM permitting both program and data to be written thereto; and

a storage device loaded with a writing control program for writing into the ROM;

wherein said ROM and said storage device are placed in different address locations in an address space of a central processing unit.

2. The microcomputer set forth in claim 1, wherein said storage device loaded with the writing control program is a mask ROM.

3. The microcomputer set forth in claim 1 or 2, wherein said storage device loaded with the writing control program is a RAM.

3. Detailed Description of the Invention

[Industrial Field of Application]

This invention relates to a microcomputer technique and, more particularly, to a technique that is effectively applied to a single-chip microcomputer incorporating an electrically

writable ROM such as an EEPROM (electrically erasable and programmable read-only memory). The invention relates to a technique that is effectively used for a microcomputer built, for example, in an IC card.

[Prior Art]

In recent years, so-called IC cards have attracted attention as alternatives to magnetic cards. Such an IC card has a built-in PROM (UV-erasable programmable ROM) in which data such as ID (identification code) is stored, whereby the card can be functioned as an identification card used instead of a key, as described, for example, in JP-B-56-19665.

The present inventor has discussed single-chip microcomputers which have built-in EEPROMs and are adapted to be incorporated in IC cards as described above. The following are techniques which are not well-known but have been discussed by the present inventor. A summary of them is as follows.

Fig. 6 shows the configuration of a microcomputer 10 discussed by the present inventor.

The microcomputer 10 shown in the figure is of the single chip type and has a built-in EEPROM. The microcomputer includes a CPU (central processing unit) 1, a RAM (random access memory) 2, a mask ROM (non-erasable memory ROM) 3, EEPROMs 41, 42, an I/O (input-output) unit 5, peripheral circuitry 6, and an EEPROM-writing control portion 7 all of which are built on the same semiconductor chip. The various portions 1-7 are

interconnected by address bus LA and data bus LD.

This single-chip microcomputer 10 is incorporated, for example, in an IC card and used. Communications of data Dx with the outside are performed all via the CPU 1 as shown in Fig. 7, which illustrates the microcomputer 10 shown in Fig. 6 while taking notice of the flow of the data Dx through the microcomputer. The microcomputer 10 can be so designed that the internal software cannot be known unless an appropriate software "key" is used. This secures the single-chip microcomputer built in the IC card.

The two EEPROMs 41 and 42 are equivalent and independent of each other. As shown in Fig. 8, one EEPROM 41 is used as a so-called user program area (M1). A program that is created by the user at will is previously written into this area. Writing of the program is done into the EEPROM 41 directly from the outside after stopping the CPU 1 under external control. This method of programming a PROM is made public, for example, by "Hitachi Microcomputer Data Book, 8-bit Single Chip", pp. 823-865, issued in August 1984, by Hitachi Ltd. This makes it unnecessary to rewrite the mask ROM during fabrication. Hence, it is possible to cope instantly with users' diverse applications. Furthermore, if a means is provided to inhibit rewriting and reading to and from this EEPROM after the EEPROM 41 has been programmed, the built-in software can be protected effectively. The other EEPROM 42 is used as a data area (M2).

Those Dx of input-output data managed by the CPU 1 which are required to be preserved are written at any time according to the need. The writing into the EEPROM 42 is done via the writing control circuit 7 that is under control of the CPU 1. Generally, the time taken to write an EEPROM is about 1,000 times as long as the average instruction execution time of CPUs. During this writing period, the EEPROM 42 is electrically disconnected from the CPU 1, making impossible both writing and reading to and from the EEPROM 42.

On the other hand, the CPU 1 executes given processing while reading out a user program Ix2 stored in the program storage EEPROM 41 in response to each instruction. When it becomes necessary during this processing to write the preserved data Dx into the data storage EEPROM 42, writing into this EEPROM 42 is performed through the EEPROM-writing control portion 7. During the execution of the processing, reference is made, whenever necessary, to a program routine (or a program module) prepared in advance as a standard program Ix1 in the mask ROM 3. For instance, the program routine is a software timer or a program for division, and useful programs are prepared in accordance with many applications. Generally, a mask ROM can be achieved in a smaller area than an EEPROM having the same capacity. Therefore, the overall size of the semiconductor chip can be reduced by utilizing the ROM 3 without storing all the programs in the EEPROM 41.

However, the overall processing is conducted in accordance with the user program written in the EEPROM 41.

The EEPROM-writing control portion 7 as shown in Fig. 7 is controlled based on a program written in one EEPROM 41, for example. At the same time, the control portion carries out a writing operation to the other EEPROM 42. The other EEPROM 42 is disconnected from the CPU 1 while the writing operation is being carried out.

In the manner described above, the microcomputer 10 is built which can immediately meet with diverse specification requirements from users and varied applications and can preserve the data Dx in the EEPROM semipermanently according to the need.

[Problem to be Solved by the Invention]

However, the present inventor has clarified that the technique described above has the following problem:

That is, the microcomputer 10 described above needs the two mutually independent EEPROMs 41 and 42 in order to write the user's program Ix2 and to store the data Dx to be preserved. If there is only one EEPROM, a read access to the EEPROM cannot be made during writing into it. Therefore, instructions to be executed by the CPU 1 cannot be read out. Accordingly, the program and the data are stored in the mutually independent EEPROMs 41 and 42 as described above. Instructions are read from one EEPROM 41. At the same time, control of writing into

the other EEPROM 42 is provided based on the instructions read out.

For this purpose, however, the two mutually independent EEPROMs 41 and 42 are necessary. Moreover, the EEPROMs 41 and 42 must have sufficient storage areas M1 and M2, respectively, to satisfy diverse specification requirements from users from various sectors. For instance, both EEPROMs 41 and 42 must cope with a large program size and a small data size. Conversely, the EEPROMs must cope with a large data size and a small program size. To satisfy both of these two requirements, it is eventually urged that the storage capacities of the EEPROMs 41 and 42 be set large. Additionally, if both EEPROMs 41 and 42 are made to have a large storage capacity, the storage capacity of any one of them is greatly wasteful and cannot be utilized effectively. This waste is likely to be incurred.

Furthermore, each of the EEPROMs 41 and 42 has a memory array and peripheral circuits such as circuits for entering and delivering data including sense amplifier and driver circuit and a circuit for selecting addresses. Therefore, where a plurality of EEPROMs are formed independently, the peripheral circuits including the sense amplifiers and drivers are provided inside the respective EEPROMs and hence a great number of circuit elements are required. Accordingly, it is urged that the overall size of each EEPROM be increased.

Accordingly, the present inventor discussed the

following method. A program for controlling the EEPROM 42 is loaded in the EEPROM 41. Data to be referenced by the program in the EEPROM 42 is also loaded in the EEPROM 41. Data to be referenced by the program in the EEPROM 41 is loaded in the EEPROM 42 together with a program for controlling the EEPROM 41. If this structure is adopted, the program storage area and the data storage area in each of the EEPROMs 41 and 42 can be made variable. In this case, the aforementioned problem regarding the memory area or size is somewhat mitigated. Even in this case, however, the EEPROMs 41 and 42 have their respective independent peripheral circuits such as sense amplifiers and decoder circuits, and thus the disadvantage regarding the overall size of each EEPROM or the size of the semiconductor chip is not fully removed.

As described so far, the aforementioned microcomputer has the advantages that it can quickly cope with various specification requirements from users and varied applications and that it can semipermanently preserve data D_x in the EEPROMs according to the need. However, there are the two independent EEPROMs 41 and 42 having sufficiently large storage capacities. Therefore, the hardware load to the construction is large. Especially, in the single chip type, the size of the semiconductor chip is unavoidably great but the efficiency of utilization of the hardware resources is not proportionately high. Presence of this problem has been clarified first by the

present inventor. Especially, where a microcomputer is built in an IC card, there is a strong demand for a reduction in size of the semiconductor chip in order to improve the card strength. Nonetheless, the semiconductor chip described above doesn't meet this requirement.

The present invention is intended to provide a technique which retains the aforementioned advantages of a microcomputer (i.e., capable of quickly complying with various specification requirements from users and varied applications and of semipermanently preserving data Dx in an EEPROM according to the need) and which permits a reduction in size of the hardware structure and enhances the efficiency of utilization of the hardware resources.

[Means for Solving the Problem]

A typical invention disclosed herein is summarily and briefly described as follows.

A microcomputer having an electrically writable ROM permitting both user program and data required to be preserved to be written thereto (i.e., the ROM is shared between program and data) and a separate storage device loaded with a writing control program for writing into the writable ROM. The ROM and storage device are placed in different address locations in an address space of a CPU.

[Operation]

According to the above-described means, when data is

written into the writable ROM, the CPU is made to jump to the separate storage device only at this time. The CPU can be made to perform given writing control processing even during writing into the writable ROM by executing a given writing control program preloaded there. Consequently, user program area and data area can be placed within one writable ROM. In addition, the ratio of the sizes of the areas can be selected at will. Thus, it is possible to retain the advantages that the microcomputer can quickly cope with various specification requirements from users and that it can semipermanently preserve the data Dx in the ROM according to the need. At the same time, the scale of the hardware construction can be reduced and the efficiency of utilization of the hardware resources can be enhanced. That is, the object is achieved.

[Embodiment]

The preferred embodiment of the present invention is hereinafter described with reference to the drawings. Note that like reference numerals indicate identical or corresponding components in the various figures.

Fig. 1 shows one embodiment of the main portion of a microcomputer 10 to which a technique according to this invention is applied.

The microcomputer 10, whose main portions are shown in the figure, is of the single chip type and has an EEPROM 4 into which user program 1x2 and data to be preserved are written

at an arbitrary proportion, in addition to a CPU 1. Additionally, the microcomputer has a so-called mask ROM 3 in which a writing control program for performing writing into the EEPROM 4 is unalterably preloaded as a part of a standard program Ix1.

The mask ROM 3 may store only the writing control program or alternatively store a standard program routine as described above. Here, the writing control program includes, for example, a program for activating the writing control circuit 7 or a program for detecting completion of writing. If the amount of data to be written is great, the data prepared in a given area within the RAM 2 may be transferred sequentially to the EEPROM 4, and writing may be performed. The EEPROM 4 and mask ROM 3 are placed in different address locations in the address space of the CPU 1. The switch shown in Fig. 1 is imaginary and used to indicate that the program executed by the CPU 1 is transferred to the mask ROM 3 by a CALL instruction when writing into the EEPROM 4 is done and that the program is returned to the EEPROM 4 by a RETURN instruction after completion of the writing.

In this case, a CALL instruction to a specific routine in the mask ROM 3 is written into the EEPROM 4 in place of the writing control program for the EEPROM 4. On the other hand, a RETURN instruction to the EEPROM 4 is written into the mask ROM 3 at the end of the writing control program together with the writing control program for the EEPROM 4.

Fig. 2 shows one embodiment of the whole construction of the microcomputer 10 shown in Fig. 1.

As shown in Fig. 2, the microcomputer 10 incorporates therein a RAM 2 providing a working area to the CPU 1, an input/output (I/O) unit 5 for exchanging data Dx with the outside, peripheral circuitry 6, and an EEPROM-writing control portion 7, in addition to the constituent elements described above (i.e., CPU 1, mask ROM 3, and EEPROM 4). Since the microcomputer includes these constituent elements therein, it is made suitable as a single-chip microcomputer to be incorporated in an IC card, for example. The various portions (1 to 7) in the microcomputer 10 are interconnected by address bus LA and data bus LD.

Control signals for the memories and peripheral circuits are omitted.

Fig. 3 shows the microcomputer shown in Fig. 2 by taking notice of the flow of data Dx through the microcomputer. As shown in the figure, exchange of data Dx with the outside is done all through the CPU 1. Consequently, the built-in software cannot be known unless a "key" formed by suitable software is used.

Fig. 4 shows three examples of the state of the address space of the CPU 1 by means of memory maps. As shown in Fig. 4, both the user program area M1 and the data area M2 are allotted in an arbitrary proportion to in the storage region M formed

by the EEPROM 4.

Fig. 5 shows a flowchart of an example of the processing operation when the CPU 1 controls writing into the EEPROM 4.

In Fig. 2, the CPU 1 reads in the program Ix2 written in the user program area M1 according to each one instruction and, at the same time, executes a given processing operation (step S6).

If it becomes necessary to write the data Dx (which is required to be saved) into the EEPROM 4 during the processing (step S1), the CPU 1 jumps to the first address of the writing control program in the standard program area Ix1 stored in the mask ROM 3 in response to a CALL instruction(step S2). The CPU 1 controls writing into the EEPROM 4 in accordance with the writing control program (step S3). Consequently, writing into the EEPROM 4 is done via the EEPROM-writing control portion 7. During this writing, the EEPROM 4 is disconnected from the CPU 1.

When the writing is subsequently completed, the CPU 1 judges the completion of the writing based on a flag generated from the writing control portion 7 or an interrupt request (step S4). Then, the CPU 1 returns from the mask ROM 3 to the program area M1 of the EEPROM 4 and resumes the reading of the user program from an address next to the address at the time of jump (step S5). The CPU 1 continues execution of the user program in the EEPROM 4 until the processing is complete or a next data

writing request is produced (step S6).

As described above, the user program area M1 and the data area M2 can be placed in the single EEPROM 4. Also, since the ratio of the sizes of both areas M1 and M2 can be selected at will, the size of the program area M1 can be set large instead of setting the size of the data area M2 small, for example, as shown in the three examples in Fig. 4, even if the size of the storage area M of the whole EEPROM is not so large. Alternatively, the size of the data area M2 is set large instead of setting the size of the program area M1 small. In this way, efficient use can be accomplished by flexible use of the storage area M.

Consequently, the object is achieved. That is, the advantages (i.e., capable of quickly complying with users' various applications and of semipermanently preserving data ~~Dx in the EEPROM 4 according to the need~~) are retained. At the same time, a reduction in the size of the hardware structure is permitted. Moreover, the efficiency of utilization of the hardware resources is enhanced.

The return to the user program upon completion of the writing into the EEPROM need not depend on a flag generated from the writing control portion 7 or on an interrupt request as in the embodiment. For example, an appropriate working register in the CPU 1 may be so configured that the register is started to be operated simultaneously with the start of the

writing into the EEPROM 4, the register is used as one kind of counter or timer updated at given intervals during the operation, and the returning operation is performed when the content of the working register reaches a given value. That is, the CPU 1 may count a forecast given writing time. On completion of this counting, completion of the writing into the EEPROM may be checked in software. In this case, the setting of the writing time and control of the subsequent returning operation may be provided in hardware by a dedicated circuit such as a timer circuit.

In the above-described embodiment, the user program is so constructed that it stops the CPU 1 under external control and directly writes into the user program area M1 of the EEPROM 4 from the outside, although this scheme does not restrict the invention.

~~This writing of the user program may be carried out in~~ the manner described below. The CPU 1 receives a program from the outside via the I/O unit 5 in accordance with the program in the mask ROM 3 and writes into the user program area M1 of the EEPROM 4 sequentially. In this example, there is no means for making direct access to the built-in EEPROM 4 from the outside and so the security protection function is strengthened. The suitability of the single-chip microcomputer built in an IC card can be increased.

In this case, a decision as to whether writing into the

user program 4 has been already done may be made by incorporating a flag within the EEPROM 4 and judging the state. According to the state of the flag, the start address used after resetting the CPU 1 may be modified.

The writable ROM is not limited to an electrically writable and erasable ROM such as an EEPROM. A UV-erasable EPROM can also be used.

In the embodiment described above, the writing is done by the writing control circuit 7 for a given time, though the present invention is not limited to this.

Generally, an EPROM has a longer writing time than an EEPROM. Therefore, in the above-described method in which the writing time is constant, the response time will increase when the microcomputer is built in an IC card. Since process variations of EPROM devices are large, their writing times are set taking account of the worst case. Therefore, in many cases, the writing time is consumed excessively.

Accordingly, the present inventor has devised a construction in which the CPU 1 can not only activate the writing control circuit 7 but also stop it. That is, a flag PGM is provided in the writing control circuit 7. When the CPU 1 sets the flag PGM, the writing is started. When the flag PGM is reset, the writing is terminated. The writing time may be measured, for example, by software as described above. Where a timer circuit is built in, this may be utilized.

Fig. 9 is a flowchart illustrating one example of the writing control program to be loaded into the mask ROM 3 in the case described above.

First, the CPU 1 sets writing address data for the EPROM, and the EEPROM latches the address data (step S1). Then, the CPU 1 clears the contents of a certain register N (step S2), adds +1 to the register N (step S3), and then sets the flag PGM (step S4). A given unit time such as 1 ms is counted (step S5), then the flag PGM is reset (step S6), and the writing for the unit time is terminated.

Thereafter, a decision is made as to whether writing has been performed correctly (step S7). This decision is carried out by reading the EPROM and comparing the read contents with the written data. The latched data may be prevented from being destroyed during the reading, though the invention is not limited to this scheme. If the result of comparison indicates a disagreement, the CPU 1 judges the value of the register N (step S₈). If it is equal to or less than 24, the CPU 1 returns to the step S3 and again performs the writing for the unit time. Where the writing operation for the unit time is repeated 25 times (i.e., $N = 25$), if a disagreement still occurs, it is determined that there is a failure (step S12), and the processing is terminated.

If the decision indicates an agreement, the CPU 1 sets the flag PGM (step S8). The CPU counts $3 \times N$ ms (step S9), and

then the CPU clears the flag PGM (step S10), terminating the processing. That is, an overwriting operation is performed for a time that is three times as long as the time N ms taken until a decision indicating an agreement is obtained.

In this way, high-speed, reliable writing conforming to the device characteristics can be accomplished. The writing time and the response time can be shortened.

The above-described method yields the advantages that it is possible to quickly comply with various specification requirements from users and varied applications and that data D_x can be semipermanently preserved in an EPROM according to the need. At the same time, the scale of the hardware structure can be reduced. In addition, the efficiency of utilization of the hardware resources can be enhanced. Moreover, the response time can be shortened.

[Advantages of the Invention]

Advantages obtained by a typical aspect of the invention disclosed herein are briefly described below.

A user program area and a data area can be placed in one EEPROM within a microcomputer having the built-in EEPROM. The ratio of the sizes of the areas can be selected at will. Thus, it is possible to quickly cope with various specification requirements from users and varied applications. Furthermore, data D_x can be preserved semipermanently in the EEPROM according to the need. The scale of the hardware structure can

be reduced while retaining the advantages described above. In addition, the efficiency of utilization of the hardware resources can be enhanced. In addition, where the invention is applied to a single-chip microcomputer to be built in an IC card, the strength of the card can be increased owing to a reduction in size of the semiconductor chip.

While the invention made by the present inventor has been described in detail based on its embodiment, the invention is not limited to the above embodiment. Obviously, various changes and modifications are possible without departing from the gist. For example, a writing control program may be previously loaded in the mask ROM 3 or EEPROM 4. When writing into the EEPROM 4 is performed, the loaded writing control program may be transferred to the RAM 2 to cause the CPU 1 to execute the program.

In the embodiment described above, the invention is applied to a microcomputer in the form of a single chip for an IC card. The invention is not limited to this. For example, the invention can also be applied to a board type microcomputer or the like.

The invention can be applied at least to microcomputers using an EEPROM in which both program and data are stored.

4. Brief Description of the Drawings

Fig. 1 is a block diagram showing main portions of a microcomputer which has a built-in EEPROM and to which a

technique according to this invention is applied;

Fig. 2 is a block diagram showing an example of the whole configuration of the microcomputer shown in Fig. 1;

Fig. 3 is a block diagram illustrating the microcomputer shown in Fig. 2 taking notice of the flow of data through the microcomputer;

Fig. 4 is an address map illustrating three states of the address space of the CPU inside the microcomputer shown in Fig. 2;

Fig. 5 is a flowchart illustrating an example of operation of the microcomputer shown in Fig. 2;

Fig. 6 is a block diagram showing the configuration of a microcomputer discussed prior to this invention, the microcomputer having a built-in EEPROM;

Fig. 7 is a block diagram illustrating the microcomputer shown in Fig. 6 taking notice of the flow of data through the microcomputer;

Fig. 8 is an address map illustrating the state of the address space of the CPU in the microcomputer shown in Fig. 6; and

Fig. 9 is a flowchart illustrating a writing control program in a case where an EPROM is built in.

CPU 1: central processing unit; RAM 2: random access memory;
mask ROM 3: non-erasable read-only memory;

EEPROMs 41, 42: electrically erasable and programmable ROMs;

I/O 5: input/output unit

⑨ 日本国特許庁(JP)

⑩ 特許出願公開

⑫ 公開特許公報(A)

昭63-266698

⑮ Int. Cl.⁴

識別記号

庁内整理番号

⑬ 公開 昭和63年(1988)11月2日

G 11 C 17/00
G 06 F 9/06
12/02
12/06

3 2 0

B-7341-5B
S-7361-5B
A-8841-5B
V-8841-5B

審査請求 未請求 発明の数 1 (全10頁)

⑭ 発明の名称 マイクロコンピュータ

⑯ 特 願 昭62-99772

⑰ 出 願 昭62(1987)4月24日

⑱ 発 明 者 三 ツ 石 直 幹 東京都小平市上水本町1450番地 株式会社日立製作所武蔵工場内

⑲ 出 願 人 株式会社日立製作所 東京都千代田区神田駿河台4丁目6番地

⑳ 代 理 人 弁理士 小川 勝男 外1名

明 細 書

1. 発明の名称

マイクロコンピュータ

2. 特許請求の範囲

1. 電気的に書込可能なROMを備えたマイクロコンピュータであって、プログラムとデータの両方が書込可能にされた電気的に書込可能なROMと、このROMに書込を行うための書込制御プログラムを格納する記憶装置とを備えるとともに、上記ROMと上記記憶装置が、中央処理ユニットのアドレス空間上にて、互いに異なるアドレス位置に配置されたことを特徴とするマイクロコンピュータ。
2. 上記書込制御プログラムを格納する記憶装置がマスクROMであることを特徴とする特許請求の範囲第1項記載のマイクロコンピュータ。
3. 上記書込制御プログラムを格納する記憶装置がRAMであることを特徴とする特許請求の範囲第1項または第2項記載のマイクロコンピュータ。

3. 発明の詳細な説明

〔産業上の利用分野〕

この発明は、マイクロ・コンピュータ技術、さらにはEEPROM(エレクトリカル・エレーサブル・アンド・プログラマブル・リード・オンリ・メモリ)のような電気的に書込が可能なROMを内蔵したシングルチップ型マイクロコンピュータに適用して有効な技術に関するもので、たとえば、ICカードに内蔵されるマイクロ・コンピュータに利用して有効な技術に関するものである。

〔従来の技術〕

最近、磁気カードなどに代わるものとして、いわゆるICカードが注目されている。このICカードは、たとえば特公昭56-19665号公報などに記載されているように、ID(識別コード)などのデータを記憶させたP-ROM(紫外線消去型のプログラマブル・ROM)を内蔵することにより、たとえばキーの代わりをなす識別カードとして機能させることができる。

ここで、本発明者は、たとえば上述したとき ICカードに内蔵するのに適したEEPROM内蔵型のシングルチップ型マイクロ・コンピュータについて検討した。以下は、公知とされた技術ではないが、本発明者によって検討された技術であり、その概要は次のとおりである。

第6図は本発明者によって検討されたマイクロ・コンピュータ10の構成を示す。

同図に示すマイクロ・コンピュータ10はEEPROM内蔵のシングルチップ型であって、CPU(中央処理ユニット)1、RAM(ランダム・アクセス・メモリ)2、マスクROM(固定記憶ROM)3、EEPROM41、42、I/O(入出力ユニット)5、周辺回路6、およびEEPROM書き込み制御部7などを同一半導体チップ内に有する。各部(1~7)はアドレスバスLAおよびデータバスLDによって相互に接続されている。

このシングルチップ型マイクロ・コンピュータ10は、たとえばICカード内に内蔵されて使用

される。そして、第7図に示すように、外部とのデータDxの授受はすべてCPU1を介して行われるようになっていいる。第7図は、第6図に示したマイクロ・コンピュータ10をデータDxの流れに着目して示したものである。このマイクロ・コンピュータ10は適切なソフトウェアによる「鍵」を使用しない限り内蔵ソフトウェアを知ることができない構成とすることができ、ICカード内に内蔵されるシングルチップ型マイクロ・コンピュータとしての適正をもたせている。

ここで、EEPROM41、42は同等のものが互いに独立して2つ設けられている。そして、第8図に示すように、一方のEEPROM41は、いわゆるユーザ・プログラム領域(M1)として利用される。ここには、ユーザが任意に作成したプログラムが予め書き込まれる。このプログラムの書き込みは外部からの制御によってCPU1を停止し、外部から直接EEPROM41に対して行われる。このようなPROMのプログラム方法は例えば(株)日立製作所昭和59年8月発行

「日立マイクロコンピュータデータブック8ビットシングルチップ」823~855頁によって公知である。これにより製造工程におけるマスクROMの書き換えが不要となり、ユーザの多様な応用に即座に応ずることができる。更にEEPROM41に対するプログラムが行われた後、このEEPROMに対する再書き込みあるいは読出しを禁止することを可能とする手段を有する構成とすれば内蔵ソフトウェアの保護として効果がある。他方のEEPROM42はデータ領域(M2)として利用される。ここには、CPU1によって管理される入出力データのうち、保存を要するデータDxが必要に応じて随時に書き込まれる。このEEPROM42に対する書き込みはCPU1によって制御される書き込み制御回路7を介して行われる。一般にEEPROMの書き込みに要する時間はCPUの平均的な命令実行時間に比べ1000倍程度であって、この書き込み期間、EEPROM42はCPU1から電気的に切離され、EEPROM42の読出し、書き込み共に不可能となっている。

他方、CPU1は、プログラム格納用EEPROM41に書き込まれたユーザ・プログラムI_x2を1命令ずつ読込みながら、所定の処理動作を実行する。そして、その処理動作の過程にて要保存データDxをデータ格納用EEPROM42に書き込む必要が生じた場合には、EEPROM書き込み制御部7を介して、そのEEPROM42への書き込みを行う。この処理動作の実行に際しては、マスクROM3に予め標準プログラムI_x1として用意されているプログラム・ルーチン(あるいはプログラム・モジュール)が適宜参照される。前記プログラムルーチンは例えばソフトウェアタイマや除算のプログラムであって、多くの用途あるいは応用によって有用なプログラムが用意されている。一般にマスクROMは同容量のEEPROMに比べて小さい面積で実現可能である。このため、全てのプログラムをEEPROM41に格納せずに前記ROM3を利用することで半導体チップ全体のサイズを縮小することが可能となっている。

しかし、全体的な処理はEEP-ROM 41に書き込まれたユーザ・プログラムに従って行われる。

第7図において示されるようなEEP-ROM書き込み制御部7は、たとえば一方のEEP-ROM 41に書き込まれたプログラムに基づく制御を受けながら、他方のEEP-ROM 42への書き込み動作を行う。この他方のEEP-ROM 42は、書き込みが行われている間、CPU 1から切り離される。

以上のようにして、ユーザの多様な仕様要求及び多様な用途に対して即座に応じられ、かつデータDxを必要に応じてEEP-ROMに半永久的に保存させることが可能なマイクロ・コンピュータ10が構成されている。

〔発明が解決しようとする問題点〕

しかしながら、上述した技術には、次のような問題点のあることが本発明者によってあきらかとされた。

すなわち、上述したマイクロ・コンピュータ

例えば、データサイズは小さくてよいがプログラムサイズは大きく、あるいはプログラムサイズは小さくてよいがデータサイズは大きく、といったような2通りの要求のいずれにも対応できるようにするためには、結局、2つのEEP-ROM 41, 42のそれぞれの記憶容量をどちらも大きくせざるを得ない。さらに、2つのEEP-ROM 41, 42の各記憶容量を両方共に大きくしても、そのどちらかは記憶容量が大きく余って有効に利用されない、という無駄が生じやすい。

なお、EEP-ROM 41, 42のそれぞれは、メモリアレイとともにセンスアンプ、ドライバ回路のようなデータ入出力のための回路やアドレスを選択するための回路からなる周辺回路をもつ。それ故に、EEP-ROMが複数個独立して形成された場合、センスアンプ、ドライバなどの周辺回路がそれぞれのEEP-ROM内に設けられることになるので、多くの回路要素が必要とされる。これに応じて、EEP-ROMの全体のサイズを大きくせざるを得なくなっている。

10では、ユーザ・プログラムI x 2を書き込むためと要保存データDxを記憶するために、互いに独立した2つのEEP-ROM 41, 42が必要となる。EEP-ROMが1つだけでは、そのEEP-ROMに書き込みを行っている間、そのEEP-ROMに対する読出アクセスができなくなってしまうからである。したがって、上述したように、プログラムとデータとをそれぞれに独立した2つのEEP-ROM 41, 42に格納させ、一方のEEP-ROM 41から命令を読出しながら、その読出した命令に基づいて他方のEEP-ROM 42の書き込み制御を実行するように構成しなければならなかった。

しかし、そのためには、互いに独立した2つのEEP-ROM 41, 42が必要であり、しかも各EEP-ROM 41, 42は、各方面のユーザからの種々多様な仕様要求に対応できるようにするために、それぞれに十分に大きな記憶領域M1, M2を用意できるものでなければならない。たと

そこで、本発明者は、EEP-ROM 41に、EEP-ROM 42の制御のためのプログラムを格納するとともにEEP-ROM 42のプログラムによって参照されるべきデータを格納し、またEEP-ROM 42に、EEP-ROM 41の制御のためのプログラムとともにEEP-ROM 41のプログラムによって参照されるデータを格納することも検討した。このようにすると、EEP-ROM 41および42のそれぞれにおけるプログラム格納エリアとデータ格納エリアとを可変にすることが可能になる。この場合、前述のようなメモリエリアもしくはサイズに関する問題は幾分緩和される。しかしながら、この場合であっても、各EEP-ROM 41および42が互いに独立的なセンスアンプやデコード回路のような周辺回路をそれぞれにもつので、EEP-ROM全体のサイズもしくは半導体チップ全体のサイズに関しての不利益は十分に除去されない。

以上のように、上述したマイクロ・コンピュータでは、ユーザの多様な仕様要求及び多様な用途

に対して即座に応じられ、かつデータDxを必要に応じてEEPROMに半永久的に保存させることができるという利点を有するものの、それぞれに十分に大きな記憶容量をもつ2つの独立したEEPROM41, 42が必要であった。このため、そのハードウェア的な構成負担が大きく、とくに、シングルチップ型のものにあつては、その半導体チップサイズがどうしても大きくなってしまい、その割にハードウェア資源の利用効率が必ずしもよくない。という問題点のあることが本発明者によってはじめてあきらかとされた。特にICカードに内蔵する場合に半導体チップサイズの縮小はカード強度の向上のために強い要求があり、上記の半導体チップはこれに反するものとなっている。

本発明の目的は、上述したマイクロ・コンピュータの利点、すなわちユーザの多様な仕様要求及び多様な用途に対して即座に応じられ、かつデータDxを必要に応じてEEPROMに半永久的に保存させることができるという利点を保持しつ

された所定の書込制御プログラムを実行させることにより、書込可能なROMへの書込動作中にもCPUに所定の書込制御処理を実行させることができる。これにより、ユーザ・プログラム領域とデータ領域とを1つの書込可能なROM内に置くことができ、さらに各領域の大きさの割合を任意に選ぶことができる。これによって、ユーザの多様な仕様要求に即座に応じられ、かつデータDxを必要に応じて上記ROMに半永久的に保存させることができるという利点を保持しつつ、そのハードウェア的な構成規模の縮小を可能にし、かつハードウェア資源の利用効率を高められるようにする、という目的が達成される。

〔実施例〕

以下、本発明の好適な実施例を図面に基づいて説明する。

なお、各図中、同一符号は同一あるいは相当部分を示す。

第1図はこの発明による技術が適用されたマイクロ・コンピュータ10の主要部における一実施

例、そのハードウェア的な構成規模の縮小を可能にし、かつハードウェア資源の利用効率を高められるようにする、という技術を提供することにある。

〔問題点を解決するための手段〕

本願において開示される発明のうち代表的なものの概要を簡単に説明すれば、下記のとおりである。

すなわち、ユーザ・プログラムと要保存データの両方の書き込みが可能とされたプログラム/データ共用の電気的に書込可能なROMと、この書込可能なROMに書込を行うための書込制御プログラムを格納する別の記憶装置とを備えるとともに、上記ROMと上記記憶装置を、CPUのアドレス空間上にて、互いに異なるアドレス位置に配置する、というものである。

〔作用〕

上記した手段によれば、書込可能なROMにデータを書き込むのに際し、そのときだけCPUを別の記憶装置へジャンプさせて、そこに予め格納

例を示す。

同図にその主要部を示すマイクロ・コンピュータ10はシングルチップ型のものであつて、CPU1とともに、ユーザ・プログラムIx2と要保存データの両方が任意の割合で書き込まれるEEPROM4を備える。これとともに、そのEEPROM4に書込を行うための書込制御プログラムが標準プログラムIx1の一部として予め固定的に格納されたいわゆるマスクROM3を備える。

マスクROM3は書込み制御プログラムのみを格納したものであつてもよく、あるいは更に前記した標準プログラムルーチンを格納したものであつてもよい。ここで、書込み制御プログラムは例えば書込み制御回路7に対する起動のためのプログラムあるいは書込みの終了を検出するためのプログラムを含むようなものである。さらに書込みデータが多量である場合にはRAM2内の所定の領域に用意されたデータを順次EEPROM4に対して転送し、書込みを行うものであつてもよい。

さらに、そのEEP-ROM 4とマスクROM 3はそれぞれ、上記CPU 1のアドレス空間上に、互いに異なるアドレス位置に配置されている。ここで、第1図におけるスイッチは仮想的なものであって、CPU 1の実行プログラムがEEP-ROM 4に対する書き込み時にはコール命令によってマスクROM 3に移り、書き込み終了後リターン命令によってEEP-ROM 4に戻ることを示すものである。

この場合、EEP-ROM 4内には、そのEEP-ROM 4への書き込み制御プログラムの代わりに、マスクROM 3内の特定ルーチンへのコール命令が書き込まれるようになっている。他方、マスクROM 3内には、EEP-ROM 4のための書き込み制御プログラムとともに、この書き込み制御プログラムの最後にEEP-ROM 4へのリターン命令が書き込まれるようになっている。

第2図は、第1図に示したマイクロ・コンピュータ10の全体的な構成の一実施例を示す。

同図に示すように、上記マイクロ・コンピュ

ータ10には、上述した構成要素すなわちCPU 1、マスクROM 3、およびEEP-ROM 4のほか、CPU 1の作業領域を提供するRAM 2、外部に対してデータDxの受け渡しを行うI/O（入出力ユニット）5、周辺回路6、およびEEP-ROM書き込み制御部7などが内蔵されている。これらを内蔵することにより、たとえばICカード内に内蔵されるシングルチップ型マイクロ・コンピュータとしての適性をもたせられている。マイクロ・コンピュータ10内の各部（1～7）はアドレスバスLAおよびデータバスLDによって相互に接続されている。

各メモリや周辺回路に対する制御信号は省略されている。

第3図は、第2図に示したマイクロ・コンピュータ10を、データDxの流れに着目して示す。同図に示すように、外部とのデータDxの授受はすべてCPU 1を介して行われるようになってい

る。これにより適切なソフトウェアによる「鍵」を使用しない限り、内蔵ソフトウェアを知ること

その書き込み制御プログラムにしたがってEEP-ROM 4の書き込み制御処理を実行する（ステップS3）。これにより、EEP-ROM書き込み制御部7を介して、そのEEP-ROM 4への書き込みが行われる。この書き込みが行われている間、EEP-ROM 4はCPU 1から切り離される。

この後、書き込みが完了すると、CPU 1は、たとえば書き込み制御部7側から発せられるフラグあるいは割込み要求に基づいて、書き込みの完了を判定する（ステップS4）。すると、CPU 1は、マスクROM 3からEEP-ROM 4のプログラム領域M1にリターンし、ジャンプ時のアドレスの次の番地からユーザ・プログラムの脱込みを再開する（ステップS5）。そして、処理の終了あるいは次のデータ書き込み要求が発生するまで、EEP-ROM 4のユーザ・プログラムを実行する（ステップS6）。

以上のようにして、ユーザ・プログラム領域M1とデータ領域M2とを1つのEEP-ROM 4内に置くことができるようになっている。これと、

ができない構成とすることが可能である。

第4図は、上記CPU 1のアドレス空間の状態の3つの例をそれぞれメモリ・マップによって示す。同図に示すように、上記EEP-ROM 4による記憶領域M内には、ユーザ・プログラム領域M1とデータ領域M2の両方が任意の割合でもって割り当てられるようになっている。

第5図は、上記CPU 1がEEP-ROM 4への書き込み制御を行う場合の処理動作例をフローチャートによって示す。

第2図において、CPU 1は、ユーザ・プログラム領域M1に書き込まれたプログラムIx2を1命令ずつ脱込みながら、所定の処理動作を実行する（ステップS6）。

ここで、その処理動作の過程にて要保存データDxをEEP-ROM 4に書き込む必要が生じると（ステップS1）、CPU 1は、コール命令によって、マスクROM 3に格納された標準プログラム領域Ix1中の書き込み制御プログラムの先頭アドレスにジャンプする（ステップS2）。そして、

ともに、両領域M1とM2の大きさの割合を任意に選ぶことができるので、EEPROM全体の記憶領域Mのサイズがそれほど大きくなくとも、たとえば第4図に3つの例を示すように、データ領域M2のサイズを小さくする代わりにプログラム領域M1のサイズを大きくとったり、あるいはプログラム領域M1のサイズを小さくする代わりにデータ領域M2のサイズを大きくとったり、といったように記憶領域Mを融通し合って効率良く利用することができる。

これによって、ユーザの多様な用途に対して即座に応じられ、かつデータDxを必要に応じてEEPROM4に半永久的に保存させることができるという利点を保持しつつ、そのハードウェア的な構成規模の縮小を可能にし、かつハードウェア資源の利用効率を高められるようにする、という目的が達成される。

ここで、EEPROMの書き込みが完了されたときのユーザプログラムへのリターンは、実施例のように書き込み制御部7から発せられるフラグあるいは

M3のプログラムに従ってCPU1がI/Oユニット5を介して外部よりプログラムを受信し、順次EEPROM4のユーザプログラム領域M1に対して書き込む構成としてもよい。この例では、内蔵EEPROM4に対して外部から直接アクセスする手段を持たないために、機密保護機能が強化され、ICカード内に内蔵されるシングルチップ型マイクロ・コンピュータとしての適正を増大させることができる。

この場合、ユーザプログラム4に対する書き込みが既に行われているか否かは、EEPROM4内にフラグを有してその状態で判定すればよい。このフラグの状態に応じて、CPU1のリセット後のスタートアドレスを変更するような構成にしてもよい。

また、書き込み可能なROMとしては、EEPROMのような電気的に書きおよび消去可能なROMだけではなく、紫外線消去型のEPROMも利用できる。

上記の例では、特に制限はされないものの、書

込みは書き込み要求によらなくてもよい。たとえば、CPU1内の適当な作業レジスタが、EEPROMへの書き込み動作の開始と同時に動作開始されて、その動作中に一定周期で更新される一種のカウンタもしくはタイマとして利用され、かかる作業レジスタの内容が所定値に達したときに上記リターン動作が実行されるように構成されてもよい。つまり、CPU1があらかじめ見込まれる所定の書き込み所要時間を計時し、この計時が完了した時点でEEPROMへの書き込み動作の完了をソフトウェア的にチェックする構成であってもよい。この場合、書き込み時間の設定とその後のリターン動作の制御は、タイマ回路のような専用回路によってハードウェア的に行わせるようにしてもよい。

上記した例では、ユーザプログラムは、特に制限されないものの、外部からの制御によってCPU1を停止し、外部から直接EEPROM4のユーザプログラム領域M1に対して書き込みを行う構成となっている。

このユーザプログラムの書き込みは、マスクROM

は書き込み制御回路7によって行われ、一定時間の書き込みが行われている。

EPROMの場合には、一般にEEPROMに比して書き込み時間が長い。このために、上記のような書き込み時間一定の方法ではICカードに内蔵した場合には応答時間の増加を招いてしまう。ここで、EPROM素子のプロセスバラツキが大きいことにより、ワーストケースを考慮して書き込み時間は設定されるために多くの場合、書き込み時間が必要以上に費されてしまっている。

そこで、本発明者は書き込み制御回路7に対してCPU1が起動をかけると共に更に、停止も可能とできる構成することを考えた。即ち、書き込み制御回路7内にフラグPGMを設け、このフラグPGMをCPU1がセットすると書き込みが開始され、前記フラグをリセットすると書き込みが終了されるというものである。書き込み時間は、例えば前記したようにソフトウェアによって計時されてもよいし、或はタイマ回路を内蔵しているものにあつては、これを利用してよい。

第9図は上記の場合のマスクROM3内に格納されるべき書き込み制御プログラムの一実施例を示すフローチャートである。

まず、CPU1がEEPROMに対する書き込みアドレス・データを設定し、EEPROMはこれらをラッチする(ステップS1)。次にCPU1は特定レジスタNの内容をクリアし(ステップS2)、上記レジスタNに+1の加算を行った(ステップS3)後にフラグPGMをセットする(ステップS4)。所定の単位時間例えば1msの計時を行い(ステップS5)、その後フラグPGMをリセットし(ステップS6)単位時間の書き込みを終了する。

この後、正しく書き込みが行われたか否かを判定する(ステップS7)。この判定は、EEPROMの読出しを行いこの読出された内容と書き込みデータを比較する。特に制限はされないが、この読出し時には前記ラッチされたデータを破壊しないように構成されている。この比較結果が不一致であればCPU1は前記レジスタNの値を判定して

(ステップS8)、24以下であれば上記ステップS3に戻り再び単位時間の書き込みを実行する。上記単位時間の書き込みが25回行われても、即ち $N=25$ となっても不一致の場合は不良と判定して(ステップS12)終了する。

上記判定の結果が一致していれば、CPU1はフラグPGMをセットし(ステップS8)、更に $3 \times N \text{ ms}$ の計時を行った(ステップS9)後に、フラグPGMをクリアして(ステップS10)終了する。即ち、上記判定結果が一致するまでに要した時間 $N \text{ ms}$ の3倍の時間による重ね書き込みが行われる。

これによって、素子特性にあった高速かつ確実な書き込みを実現することができ、書き込み時間の短縮、更に応答時間の短縮を図ることができる。

上記した方法によって、ユーザの多様な仕様要求及び多様な用途に対して即座に応じられ、かつデータDxを必要に応じてEEPROMに半永久的に保存させることができるという利点を有しつつ、そのハードウェア的な構成規模の縮小を可能

にし、かつハードウェア資源の利用効率を高められるようにし、更に応答時間を短縮することができる。

〔発明の効果〕

本願において開示される発明のうち代表的なものによって得られる効果を簡単に説明すれば、下記のとおりである。

すなわち、EEPROM内蔵型のマイクロコンピュータにあって、ユーザ・プログラム領域とデータ領域とを1つのEEPROM内に置くことができ、さらに各領域の大きさの割合を任意に選ぶことができ、これによって、ユーザの多様な仕様要求及び多様な用途に対して即座に応じられ、かつデータDxを必要に応じてEEPROMに半永久的に保存させることができるという利点を保持しつつ、そのハードウェア的な構成規模の縮小を可能にし、かつハードウェア資源の利用効率を高めることができる、という効果が得られる。さらにICカード内に内蔵すべきシングルチップ型マイクロコンピュータに適用する場合には、

半導体チップのサイズ縮小によりカードの強度を強化できるという効果が得られる。

以上、本発明者によってなされた発明を実施例にもとづき具体的に説明したが、本発明は上記実施例に限定されるものではなく、その要旨を逸脱しない範囲で種々変更可能であることはいうまでもない。たとえば、書き込み制御プログラムをマスクROM3あるいはEEPROM4に予め格納し、EEPROM4の書き込み動作を行うときに、その格納された書き込み制御プログラムをRAM2へ転送してCPU1に実行させるような構成でもよい。

また、ICカード用シングルチップ型マイクロコンピュータに適用した場合について説明したが、それに限定されるものではなく、たとえば、ボード型のマイクロコンピュータなどにも適用できる。

少なくとも、EEPROMにプログラムとデータの両方を記憶させる条件のものには適用できる。

4. 図面の簡単な説明

第1図はこの発明による技術が適用されたEEPROM内蔵型マイクロコンピュータの主要部を示すブロック図。

第2図は第1図に示したマイクロコンピュータの全体的な構成例を示すブロック図。

第3図は第2図に示したマイクロコンピュータをデータの流れに着目して示すブロック図。

第4図は第2図に示したマイクロコンピュータ内CPUのアドレス空間の3つの状態を例示するアドレスマップ。

第5図は第2図に示したマイクロコンピュータの動作例を示すフローチャート。

第6図はこの発明に先立って検討されたEEPROM内蔵型マイクロコンピュータの構成を示すブロック図。

第7図は第6図に示したマイクロコンピュータをデータの流れに着目して示すブロック図。

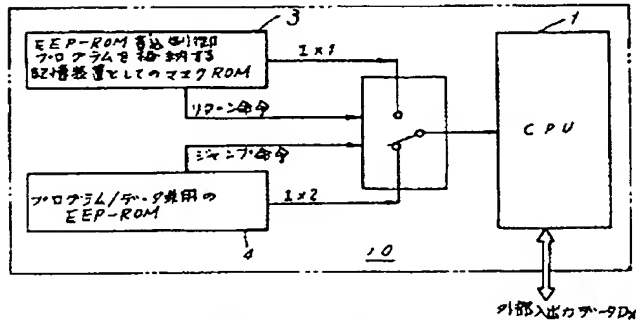
第8図は第6図に示したマイクロコンピュータ内CPUのアドレス空間の状態を例示するアドレスマップである。

第9図はEEPROMを内蔵した場合の書き込み制御プログラムを示すフローチャートである。

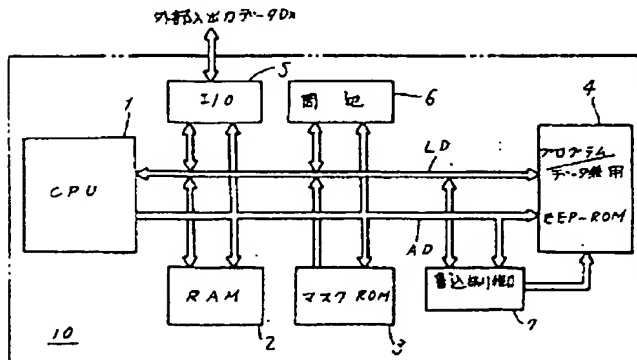
CPU 1…中央処理ユニット、RAM 2…ランダム・アクセス・メモリ、マスクROM 3…固定記憶リード・オンリ・メモリ、EEPROM 4 1、4 2…エレクトリカリ・エレーサブル・アンド・プログラマブルROM、I/O 5…入出力ユニット。

代理人 弁理士 小川 勝 男

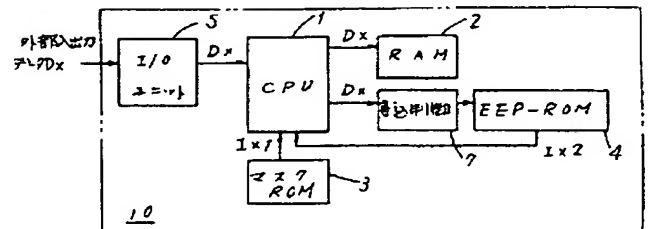
第 1 図



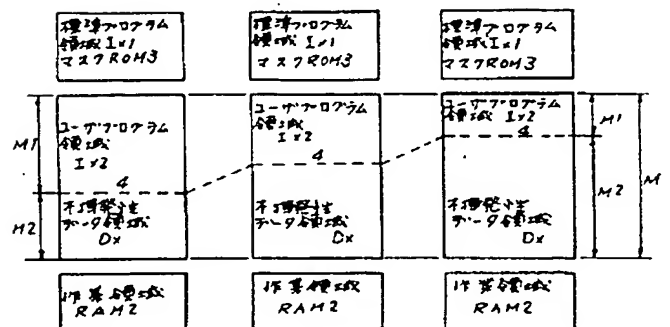
第 2 図



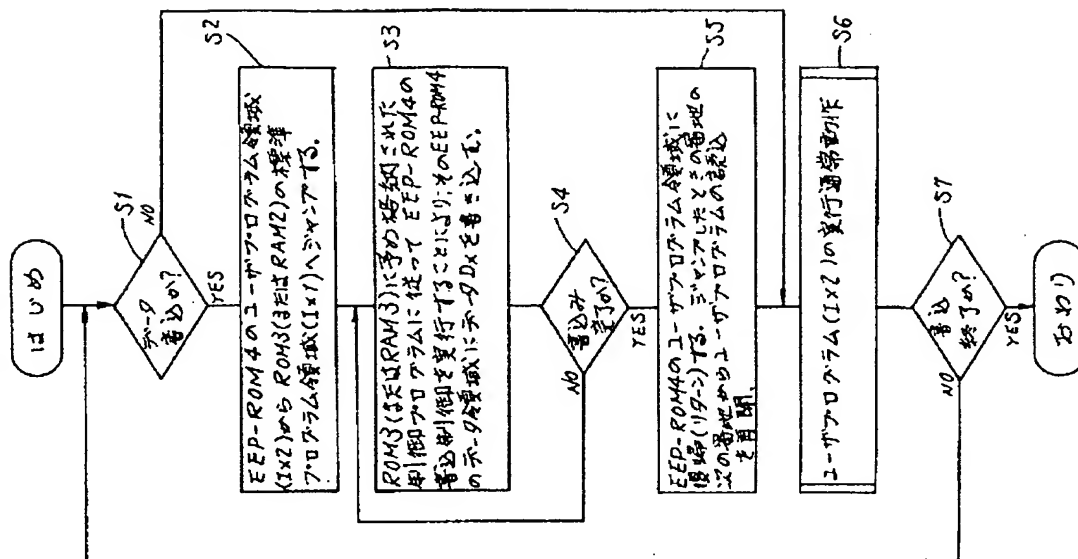
第 3 図



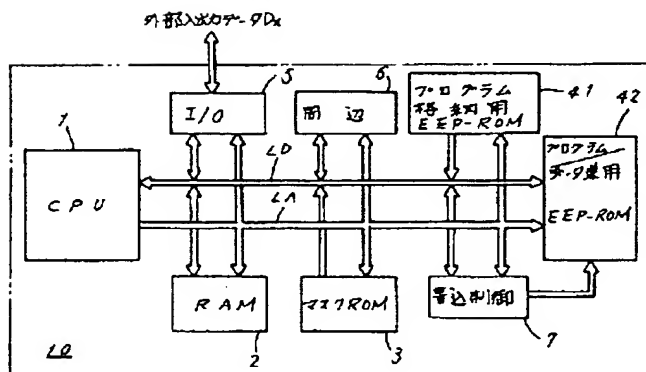
第 4 図



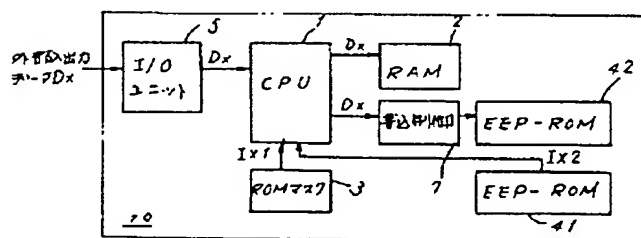
第 5 図



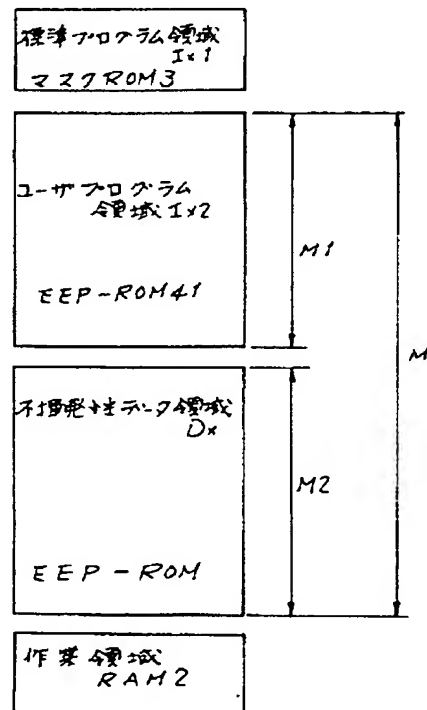
第 6 図



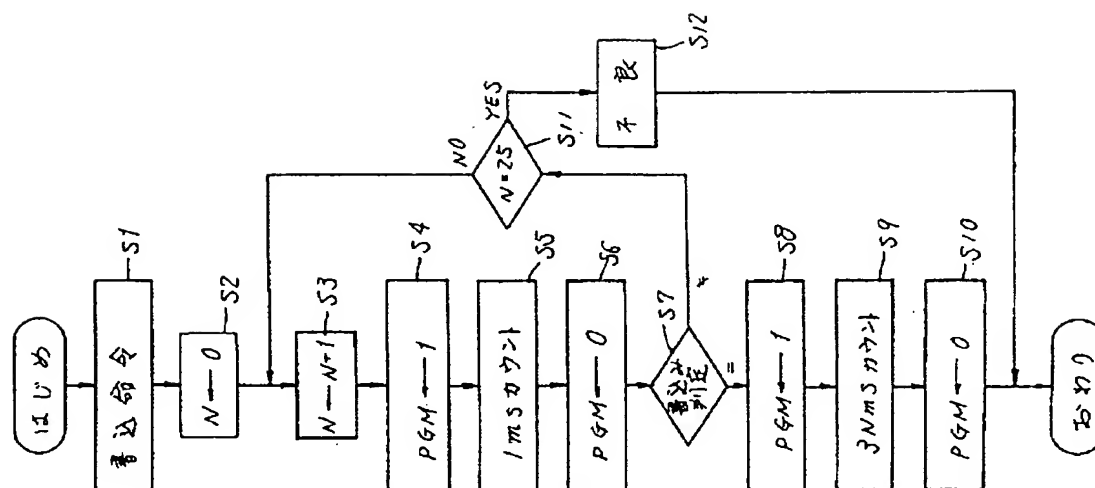
第 7 図



第 8 図



第 9 図



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☒ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☒ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.